An Assessment of Features Related to Phishing Websites using an Automated Technique

Rami M. Mohammad
School of Computing and Engineering
University of Huddersfield
Huddersfield,UK
rami.mohammad@hud.ac.uk

Fadi Thabtah
School of MIS
Philadelphia University
Amman, Jordan
ffayez@philadelphia.edu.jo

Lee McCluskey
School of Computing and Engineering
University of Huddersfield
Huddersfield, UK
t.l.mccluskey@hud.ac.uk

Abstract—Corporations that offer online trading can achieve a competitive edge by serving worldwide clients. Nevertheless, online trading faces many obstacles such as the unsecured money orders. Phishing is considered a form of internet crime that is defined as the art of mimicking a website of an honest enterprise aiming to acquire confidential information such as usernames, passwords and social security number. There are some characteristics that distinguish phishing websites from legitimate ones such as long URL, IP address in URL, adding prefix and suffix to domain and request URL, etc. In this paper, we explore important features that are automatically extracted from websites using a new tool instead of relying on an experienced human in the extraction process and then judge on the features importance in deciding website legitimacy. Our research aims to develop a group of features that have been shown to be sound and effective in predicting phishing websites and to extract those features according to new scientific precise rules.

Keywords- Website features, Phishing, Security, Rule, features extraction.

I. INTRODUCTION

Phishing attacks usually aim to acquire confidential information such as usernames, passwords and financial IDs by fooling users. Phishing attacks typically start by sending an email that appears to come from legitimate company to victims asking them to update or validate their information by visiting a link within the email. Phishers are now employing different techniques in creating websites to fool the users and tempting them, but they all use a set of common features to design phishing websites. This is since, without these features they lose the advantage of deception [1]. In general, two approaches are employed in identifying the phishing website. The first one is based on blacklist [2], by comparing the requested URL with those in that list. One drawback of this approach is that the blacklist usually cannot cover all phishing websites since within seconds a new fraudulent website is expected to be launched. The second approach is called heuristic-based, where several features are gathered from the website to classify it either phishy or legitimate. In contrast to the blacklist method, a heuristics-based solution can identify newly created phishing websites in real-time. The efficiency of heuristic-based method depends on selecting a set of discriminative features that could help distinguishing phishing websites from legitimate ones. Features can be extracted in several ways one of which is manual extraction where users derive features and judge on the website legitimacy. But users have to spend a lot of time studying the latest phishing techniques in order to be up to date with new deception methods which is hard for the majority of internet users. The second method employed in extracting phishing features is automatic extraction. This is accomplished by analyzing the webpage and extracting a set of patterns used by phishers. The techniques for analyzing the webpages involve examining its properties and all its features and patterns. Webpage properties are typically derived and extracted from HTML tags, URL address and Javascript source code [2] [3] [4]. Several studies were conducted about phishing features and their effectiveness in the process of predicting the type of websites, but these studies lack in defining precise rules to extract the features. In other words, most of the rules defined about the phishing features are only based on human experience rather than scientific experiments. This paper differs from previous research works by proposing a group of features that can be extracted automatically using our own software tool and depending on newly proposed rules that have been developed experimentally. Motivation behind the development of a set of rules to automatically extract phishing features is to reduce the false negative rate which means "classifying phishing website as legitimate". Moreover, we want to show that extracting feature automatically would be faster than manual extraction, which in turn would increase the dataset size and that allow us to conduct more experiments. Thus, improving the accuracy of our rules, or even adding some other rules.

In this article, we try to answer the following research questions:

1- What are the effective minimal sets of features that can be utilized in predicting phishing?
2- Can we suggest new rules for automatically extracting features?

This article is organized as follows: Sections II and III discuss related works and compare different phishing extraction methods presented in the literature. Section IV describes the tools we have used to extract the features and Section V introduces the structure of the proposed phishing features. Finally, section VIII measures the significance of the proposed features in detecting phishing website. We conclude in Section IX.

II. RELATED WORKS

The accuracy of predicting the type of the website necessarily depends on the extracted features goodness, which has been used in the decision process. Now, since most users feel safe against

phishing attacks if they utilize an anti-phishing tool, this throws a great responsibility for the anti-phishing tools to be accurate in predicting phishing. In this context, we believe that developing rules of thump to extracting features from website is the key to success in this issue. In this section, we review current anti-phishing approaches and the features they use in developing solutions. One approach that was proposed in [5] is a client-side defence framework, by developing SpoofGuard (plug-in). This is an open source tool [6] that examines the requested webpage and notifies the user when a spoof attack is taken place. The spoof index is calculated, if the index goes above a level specified by the user SpoofGuard warns the user of possible attack. Another approach proposed in [3], suggests a way to detect phishing websites by capturing abnormal behaviours demonstrated in these websites. Structured website consists of W3C DOM features [7]. The authors have selected six structural features as shown in Table I. After conducting experimentation, the results showed that the classifier efficiency depends on "Identity Extractor". Furthermore, the accuracy in this method was 84% which is relatively considered low.

TABLE I. ANOMALY BASED FEATURES

Feature	Feature Clarification
Abnormal URL	The hostname does not match its claimed identity.
Abnormal DNS record	No record founded in the WHOIS database for the domain.
Abnormal Anchors	In a legitimate website anchors point to same domain.
Server Form Handler	Information's are not processed on the same domain
Abnormal cookie	Cookies conflicts with website identity.
Abnormal certificate in SSL	Distinguished Names (DN) within the certificates conflicts with the claimed identities.

A literature survey on Voice Phishing "vishing" and SMS Phishing "smishing" utilizing intelligent tools and awareness security programs was given in [8]. The authors analyzed 600 phishing emails, and they collected a set of features, those are shown in Table II. They suggested a combination between the human and a proper utilization tool to derive better results in preventing phishing attack. The experiments conducted against the email dataset showed that 22% of the emails were classified as suspicious and 78% were classified as phishing. While 95% of legitimate emails were classified as non-phishing emails and 5% were classified as suspicious. The authors also conducted an experiment aiming to evaluate which feature set is more effective in predicting type of emails, the results showed that the source code features (IP based URL and Non matching URLs, Contain scripts, Number of domains) are more significant in predicting phishing emails than content features (Generic salutation, Security promises, Require a fast response, Links to https://domain).

TABLE II. AWARENESS PROGRAM FEATURES

Layer	Feature	Feature Clarification
Source code features (Front End)	IP based URL and Non-matching URLs	If the domain name has an IP address
	Contain scripts	Using onMouseOver scripts to hide or show fake URL.
	Number of Domains	Phishing websites uses multiple domains
Content Features (Back End)	Generic Salutation	Non-personalization of greeting increases the phishing possibility.
	Security promises	Phishers claims providing good security.
	Requires a fast response	To collect information before the website is turned off
	Links to https://domain	The phishers forward the victim to unsecured link.

III. DISCUSSION

Extracting features is considered the first step toward judging on the website legitimacy. Features extraction may be achieved manually, but the human factor bears the burden in extracting such features, and that would increase the false negative rate. The human methods may also increase the likelihood of exposure to phishing attack, for many reasons such as [9]:

1. Users have no idea about how computer systems work.
2. Ignoring security alerts.
3. Good visual tricks.
4. Some users do not distract themselves from the primary activities toward extracting phishing features because they believe that security system should provide such task.

These four reasons were the motivation for us to develop precise rules to extract features automatically, so increasing the proportion of predicting phishing. As if we compare between the results of detecting phishing based on human factor features with that of automatically extracted features, we note that automatically extracted features results are more accurate since research's reviewed in Section II clearly support this assumption. The problem in [5] is that it is based on superficial characteristics of current phishing attacks. Some heuristic rules are often bypassed by subtle attacks. Some methods were mentioned in the same paper to fool the password and image checks. Moreover, without wilfulness circumventing SpoofGuard's detection rules, some phishing websites are not detected by SpoofGuard. Also, warning user of attack depends mainly on the indexes selected by the user himself (i.e. human factor). In [3], capturing abnormal behaviours use features that can be extracted from the website depending on W3C DOM objects [7]. The results showed that the performance of page classifier relays on "Identity Extractor". However, this approach ignores important features that can play a key role in determining the legitimacy of website and focused only on structural features (website objects or properties). This explains the low detection rate. One solution to improve the detection rate could be using additional features. Though, this approach does not depend on any previous knowledge of the user

or experience in computer security. In [8], the responsibility in detecting online attacks is a joint effort between human elements and computerized tools. The results showed that source code features (automatically extracted) have a higher impact in predicting threats than content-based features (human based).

IV. Preparing for Feature Selection

To conduct the experiments of producing new rules related to phishing features some preparatory steps must be implemented as follows:

- Dataset preparation

A set of phishing and legitimate websites were gathered from Phishtank archive [10]. We collected 2500 phishing URLs.

- Address bar features

A JavaScript program was built to extract all features related to address bar.

- Abnormal based features

A simple PHP script for extracting those features was developed since these features deal with servers, and requires connecting to external domains such as WHOIS database [11].

- HTML and JavaScript based features

A JavaScript program was built to extract these features.

- Domain based features

These features can be extracted from WHOIS database [11], and from Alexa.com [12]. Further, we developed a PHP script to extract these features as shown in Fig 1.

V. The Proposed Computerized Based Features

Going back to the approaches presented in Section II, and after reviewing surveys, we were able to suggest the phishing prediction structure shown in Fig 2. This structure depends on extracting features from the webpage itself rather than user experience. The proposed structure consists of four main classes in which features were classified and placed in the appropriate class. First, we examine whether a page contains any text fields, since a phishing webpage requires users to input credentials through those fields [4]. If a page has at least one text input then we proceed to extract the other features. Otherwise, the extraction process is terminated. To measure the feature significance in detecting phishing we have collected 2500 datasets from the PhishTank using our tool and computed each feature frequency within the data set in order to reflect the feature importance. So, in the next section, every feature will be associated with a weight corresponding to the ratio of that feature in the data collection. These frequencies will give us an initial indication of how influential is the feature in detecting phishing websites.

A. Address Bar based Features

1. Using the IP Address

If IP address is used instead of domain name in the URL e.g. 125.98.3.123 the user can almost be sure someone is trying to steal his personal information. Sometimes the IP address is transformed to hexadecimal as in the following link: http://0x58.0xCC.0xCA.0x62/2/paypal.ca/index.html

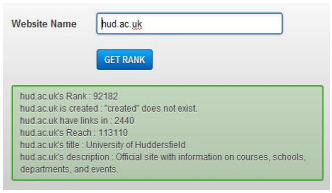

Figure 1 Result of Alexa Query

This feature is almost used in all previous studies, which give us an indication about its importance, and by reviewing our dataset we were able to find 570 URLs having IP address which constitute 22.8% of the dataset. To produce a rule for extracting this feature, we examine the domain part of the URL which lies between // and /, as shown in Fig 3. If the domain part has IP address then "True" value is assigned otherwise "False" value will be given to this feature.

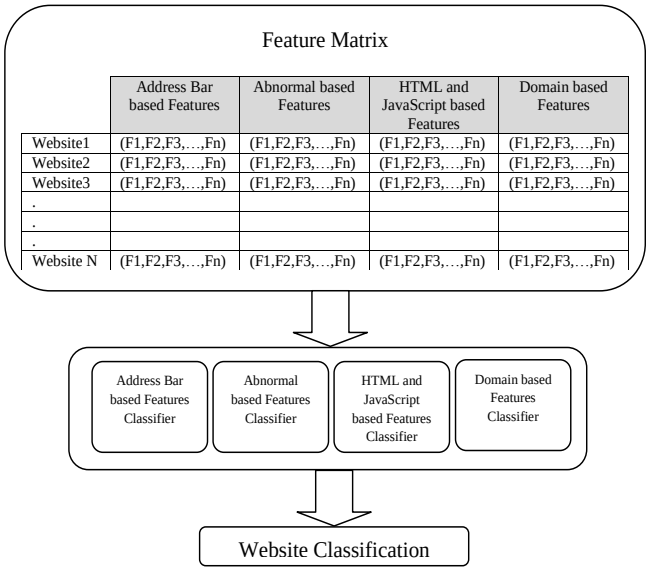

Figure 2 the proposed phishing prediction hierarchical model

Rule:

$$\text{IF}\begin{cases}\text{IP address exist in URL} \rightarrow \text{feature} = \text{True} \\ \text{otherwise} \rightarrow \text{feature} = \text{False}\end{cases}$$

2. Long URL to Hide the Suspicious Part

Phishers can use long URL's to hide the doubtful part in the address bar. Scientifically, there is no reliable length distinguishes phishing URLs from legitimate ones. As in [4], the proposed length of legitimate URLs is 75 characters or less, but the authors did not justify the reason behind their value. To ensure accuracy of our study, we calculated the length of URLs in the dataset and produced an average URL length. The results showed that if the length of the URL is greater than or equal 54 characters then the URL classified as phishing. By reviewing our

dataset we were able to find 1220 URLs lengths equals to 54 or more which constitute 48.8% of the total dataset size.

Rule:

$$\text{IF}\begin{cases}\text{URL length} < 54 \rightarrow \text{feature} = \text{NotLong}\\ \text{else if URL length} \geq 54 \text{ and } \leq 75 \rightarrow \text{feature} = \text{Suspecious}\\ \text{otherwise} \rightarrow \text{feature} = \text{VeryLong}\end{cases}$$

We have been able to update this feature rule by using a method based on frequency and thus improving upon its accuracy.

3. URL's having @ Symbol

Phishers use tricks to give the impression that the URL is legitimate using @ symbol in the URL. The browser might ignore everything prior the @ symbol since the real address often follows the @ symbol. After reviewing our dataset, we were able to find 90 URLs having @ symbol which constitute only 3.6% of the dataset. Therefore, this feature is not of high significant since its presence is rare in the dataset.

Rule:

$$\text{IF}\begin{cases}\text{URL having @ sumbol} \rightarrow \text{feature} = \text{True}\\ \text{otherwise} \rightarrow \text{feature} = \text{False}\end{cases}$$

4. Adding Prefix or Suffix Separated by (-) to Domain

Dash is rarely used in legitimate URL. Phishers resort to add suffixes or prefixes separated by (-) to the domain name. So that users feel they are dealing with the legitimate webpage. To produce a rule for this feature, we checked the frequency of the URL's in the dataset containing (-) symbol. There was 661 URLs having (-) symbol which constitute 26.4%.

Rule:

$$\text{IF}\begin{cases}\text{domain part includes } (-) \text{ symbol} \rightarrow \text{feature} = \text{True}\\ \text{otherwise} \rightarrow \text{feature} = \text{False}\end{cases}$$

5. Sub Domain and Multi Sub Domain

Assume that we have the following link http://www.hud.ac.uk/students/. A domain name always includes the top-level domain (TLD), which in our example is "uk." The "ac" part is shorthand for academic and combined ".ac.uk" is called a second-level domain (SLD) and "hud" is the actual name of the domain. Thus, we note that the legitimate URL link has two dots in the URL since we can ignore typing www. To produce a rule for extracting this feature, we have first to extract (www.) from the URL and then count the dot's in which if the number of dot's is equal to three then the URL is classified as "suspicious" since it has one sub domain. However, if the dots are greater than three it is classified as "phishing" since it will have multiple sub domains. The dataset contains 1109 URLs having three or more dots in domain part which constitute 44.4% of the dataset.

Rule:

$$\text{IF}\begin{cases}\text{dots in domain part} < 3 \rightarrow \text{feature} = \text{Low}\\ \text{else if dots in domain part} = 3 \rightarrow \text{feature} = \text{Suspecious}\\ \text{otherwise} \rightarrow \text{feature} = \text{High}\end{cases}$$

6. HTTPS (Hyper Text Transfer Protocol with Secure Sockets Layer) and SSL (Secure Sockets Layer)

HTTPS is HTTP plus SSL. You need a certificate to use any protocol that employs SSL. Legitimate websites utilize secure domain names every time sensitive information must be transferred. The existence of HTTPS is very important in giving the impression of website legitimacy, but it is not enough, since in 2005 Netcraft Toolbar Community has recognized more than 450 phishing URLs using "https" [13]. So we further check the certificate assigned with https including the extent of trust of certificate issuer, and the certificate age. Certificate Authorities that are consistently listed among the top names for trust include [14]: GeoTrust, GoDaddy, Network Solutions, Thawte, and VeriSign. By reviewing our dataset we were able to find 2321 URLs does not support HTTPS or use a fake https which constitute 92.8% of the dataset. Unlike some previous researches which consider fake HTTPS as a valid without checking the certificate authority provider our feature consider these HTTPS providers.

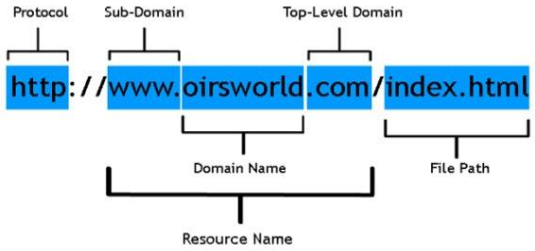

Figure 3 URL Anatomy

Rule:

$$\text{IF}\begin{cases}\text{use https issuer is trusted age} \geq 2 \text{ years} \rightarrow \text{feature} = \text{Low}\\ \text{else if}\\ \text{using Https and issuer is not trusted} \rightarrow \text{feature} = \text{Moderate}\\ \text{otherwise} \rightarrow \text{feature} = \text{High}\end{cases}$$

B. Abnormal Based Features

1. Request URL

External objects such as images within a webpage are loaded from another Domain [3]. For legitimate websites, most of objects within the webpage are linked to the same domain. For example, if the URL typed in address bar was http://www.hud.ac.uk/students/, we extract the keyword <src=> from the website source code and check whether the domain in the URL is different from that in <src>. If the result is true the website is classified as "phishing". To develop a rule for this feature, we calculate the rate of URLs in source code of the

website that have different domain than the domain typed at the address bar. If the rate is less than 20% the website is considered "legitimate" else if the rate is between 20% and 50% then the website considered "suspicious". Otherwise the website is considered "phishy". Our dataset contains 2500 URLs having this feature which constitute 100 % of the dataset. This result reveals the importance of this feature in detecting phishing.

Rule:

$$\text{IF}\begin{cases} \text{request URL \%} < 20\% \rightarrow \text{feature} = \text{Legitimate} \\ \text{else if} \\ \text{request URL\%} \geq 20\% \text{ and } 50\% \rightarrow \text{feature} = \text{Suspecious} \\ \text{otherwise} \rightarrow \text{feature} = \text{Phishy} \end{cases}$$

2. URL of Anchor

An anchor is an element defined by the <a> tag. We check (1) whether the domain of anchor is different from that of the website, and if so the website is classified as phishing. This is similar to request URL feature.

(2) If the anchor does not link to any webpage, e.g.:

<a href="#">,

<a href="#content">,

<a href="#skip">,

<a href="JavaScript ::void(0)">

Then the website is classified as phishing. By reviewing our dataset, we were able to find 581 URLs having this feature which constitute 23.2%.

Rule:

$$\text{IF}\begin{cases} \text{URL of anchor \%} < 20\% \rightarrow \text{feature} = \text{Low} \\ \text{else if} \\ \text{URL of anchor \%} \geq 20\% \text{ and} \leq 50\% \rightarrow \text{feature} = \text{Suspecious} \\ \text{otherwise} \rightarrow \text{feature} = \text{High} \end{cases}$$

3. Server Form Handler

SFH that contains empty string or "about:blank" are considered doubtful since an action should be taken upon submitted information. Furthermore, if the domain name in SFH-s is different than the domain of the webpage this gives an indication that the webpage is suspicious because the form is rarely handled by external domain server. By checking our dataset we were able to find 101 URLs having SFHs which constitute only 4.0% of the dataset size

Rule:

$$\text{IF}\begin{cases} \text{SFH is "about:blamk" or an empty string} \rightarrow \text{feature} = \text{High} \\ \text{else if SFH refers to different domain} \rightarrow \text{feature} = \text{Suspecious} \\ \text{otherwise} \rightarrow \text{feature} = \text{Low} \end{cases}$$

4. Abnormal URL

This feature can be extracted from WHOIS database [11] when the host name in URL does not match its claimed identity. For a legitimate websites, identity is typically part of its URL. 412 URLs having this feature founded in our dataset which constitute 16.4%.

Rule:

$$\text{IF}\begin{cases} \text{the host name is not included in URL} \rightarrow \text{feature} = \text{High} \\ \text{otherwise} \rightarrow \text{feature} = \text{Low} \end{cases}$$

C. HTML and JavaScript based Features:

1. Redirect Page

Open redirects found on web sites are liable to be exploited by phishers to create a link to their site. This makes the link look genuine, as it appears legitimate web site and acceptable if the site is served using SSL. When a user clicks on the link, he may be unaware that he is redirected to the phishing site. 249 URLs having redirect page were encountered in our dataset which constitute 10%.

Rule:

$$\text{IF}\begin{cases} \text{redirect page \#} \leq 2 \rightarrow \text{feature} = \text{Low} \\ \text{else if redirect page \#} > 2 \; and < 4 \rightarrow \text{feature} = \text{Suspecious} \\ \text{otherwise} \rightarrow \text{feature} = \text{High} \end{cases}$$

2. Using onMouseOver to Hide the Link

Phishers may use JavaScript to display a fake URL in the status bar to the users. 496 URLs having this feature were founded in our dataset which constitute 20%.

Rule:

$$\text{IF}\begin{cases} \text{onmouseover change the status bar} \rightarrow \text{feature} = \text{High} \\ \text{else if} \\ \text{it doesn}'\text{t change status bar} \rightarrow \text{feature} = \text{Suspecious} \\ \text{otherwise} \rightarrow \text{feature} = \text{Low} \end{cases}$$

3. Disabling Right Click

Phishers use JavaScript to disable the right click function, so that users cannot view and save the source code. 40 URLs were founded in which the right click is disabled in our dataset which constitute 1.6% of the entire dataset.

Rule:

$$\text{IF}\begin{cases} \text{right click disabled} \rightarrow \text{feature} = \text{High} \\ \text{else if right click showing an alert} \rightarrow \text{feature} = \text{Suspecious} \\ \text{otherwise} \rightarrow \text{feature} = \text{Low} \end{cases}$$

4. Using PopUp Window

It's unusual to find a legitimate website asking users to submit their credentials through a popup window. 227 URLs were founded in our dataset in which the users credential submitted through a popup window which constitutes 9.1%

Rule:

$$\text{IF}\begin{cases} \text{popup \#} \leq 2 \rightarrow \text{feature} = \text{Low} \\ \text{else if popup \#} > 2 \; and < 4 \rightarrow \text{feature} = \text{Suspecious} \\ \text{otherwise} \rightarrow \text{feature} = \text{High} \end{cases}$$

D. Domain based Features

1. Age of Domain

This feature can be extracted from WHOIS database [11]. If the domain created less than one year it is classified as "phishing", else if the domain age is more than one year and less than 2 years then it's classified as "suspicious" otherwise the website is considered "legitimate". A PHP script was created to connect to WHOIS database [11] and make a query about the domain age. 2392 URLs created less than 12 months or will expire within the coming 3 months which constitute 95.6%.

Rule:

$$\text{IF}\begin{cases}\text{age of domain is} \geq 2 \text{ years} \rightarrow \text{feature} = \text{Low} \\ \text{else if age of domain is} \geq 1 \text{ and} < 2 \rightarrow \text{feature} = \text{Suspecious} \\ \text{otherwise} \rightarrow \text{feature} = \text{High}\end{cases}$$

2. DNS Record

This feature can be extracted from WHOIS database [11]. For phishing sites, either the claimed identity in not recognized by WHOIS database or the record of the hostname is not founded [3]. If the DNS record is empty or not found then the website is classified as "phishing", otherwise it's classified as legitimate. 160 URL were found in our dataset where the DNS record is not found, and that constitute 6.4%.

Rule:

$$\text{IF}\begin{cases}\text{there are no DNS record for the domain} \rightarrow \text{feature} = \text{Phishing} \\ \text{otherwise} \rightarrow \text{feature} = \text{Legitimate}\end{cases}$$

3. Website Traffic

This feature can be extracted from Alexa database [12]. If the domain has no traffic or not being recognized by Alexa database it is classified as "phishing" otherwise if the website ranked among the top 100,000 it's classified as "legitimate" else it's classified as "Phishing". This feature has not been used before in any previous study. When reviewing our dataset we noticed that this feature has a high significant in predicting phishing websites, since it can measure the popularity of the URL. To extract this feature a PHP script has been developed to connect to Alexa server and make a query about the popularity of the URL, the result we get from our query is as shown in Fig 1. This feature constitutes 89.2% of the dataset since it appears 2331 times.

Rule:

$$\text{IF}\begin{cases}\text{webpage rank} < 100{,}000 \rightarrow \text{feature} = \text{Legitimate} \\ \text{The Ranking} > 100{,}000 \rightarrow \text{feature} = \text{Suspecious} \\ \text{otherwise} \rightarrow \text{feature} = \text{Phishing}\end{cases}$$

VI. Conclusion

This article groups features that are effective in detecting phishing websites. These features are automatically extracted without any intervention from the users and using computerized developed tool. We managed to collect and analyze 17 different features that distinguish phishing website from legitimate ones. Then, we developed a new rule for each feature. These rules can be useful in applications related to discovering phishing websites based on features. The process of extracting features using our tool is much faster and reliable than manual extraction and thus, the size of the dataset increases dramatically and that allow us to study large number of phishing pages and legitimate. After calculating the frequency for each feature, the results showed that "Request URL" has the highest significant in detecting phishing websites since it is presented in all dataset case, followed by "Age of Domain" which presented in 2392 dataset case. The next significant feature is "HTTPS and SSL" feature with frequency of 92.8%. The lowest significant feature in distinguishing phishing website is "Disabling Right Click" feature which has only appeared forty times, followed by "URL having @ symbol" feature which constituted 3.6% of the dataset size. In near future, several experiments will be conducted using data mining algorithms to extract new hidden rules concerning phishing and then modify the existing rules or adding new rules, if necessary.

References

[1] L. James , Phishing Exposed, Syngress Publishing, 2005.

[2] Nuttapong Sanglerdsinlapachai and Arnon Rungsawang, "Using Domain Top-page Similarity Feature in Machine Learning-based Web," in Third International Conference on Knowledge Discovery and Data Mining, 2010.

[3] Y. Pan and X. Ding, "Anomaly Based Web Phishing Page Detection," in In ACSAC '06: Proceedings of the 22nd Annual Computer Security Applications Conference., Dec. 2006.

[4] R. B. Basnet, A. H. Sung and Q. Liu, "Rule-Based Phishing Attack Detection," in Proceedings of the International Conference on Security and Management-SAM'11, Las Vegas, NV, USA, 2011.

[5] Neil Chou, Robert Ledesma, Yuka Teraguchi, Dan Boneh and John C. Mitchell, "Client–side defense against web–based identity theft," in 11th Annual Network and Distributed System Security Symposium (NDSS '04), San Diego, February, 2004..

[6] "SpoofGuard," [Online]. Available: http://crypto.stanford.edu/SpoofGuard/download.html. [Accessed 16 January 2012].

[7] "W3C," [Online]. Available: http://www.w3.org/TR/DOM-Level-2-HTML/. [Accessed 17 February 2012].

[8] O. Salem, H. Alamgir and K. M, "Awareness Program and AI based Tool to Reduce Risk of Phishing Attacks," in Computer and Information Technology (CIT),2010 IEEE 10th International Conference., June 29 2010-July 1 2010.

[9] S. E. Schechter, R. Dhamija, A. Ozment and I. Fischer, "The Emperor's New Security Indicators," in Proceedings of the 2007 IEEE Symposium on Security and Privacy, Washington, DC, USA, 2007.

[10] "PhishTank," October 2006. [Online]. Available: http://www.phishtank.com/. [Accessed 25 November 2011].

[11] "WhoIS," [Online]. Available: http://who.is/. [Accessed 13 March 2012].

[12] "Alexa the Web Information Company," [Online]. Available: http://www.alexa.com/. [Accessed 26 January 2012].

[13] "More than 450 Phishing Attacks Used SSL in 2005," [Online]. Available: http://news.netcraft.com/archives/2005/12/28/more_than_450_phishing_attacks_used_ssl_in_2005.html. [Accessed 8 March 2012].

[14] "Best SSL Certificates," [Online]. Available: http://www.bestsslcertificates.com/articles27.html. [Accessed 8 March 2012].